

---

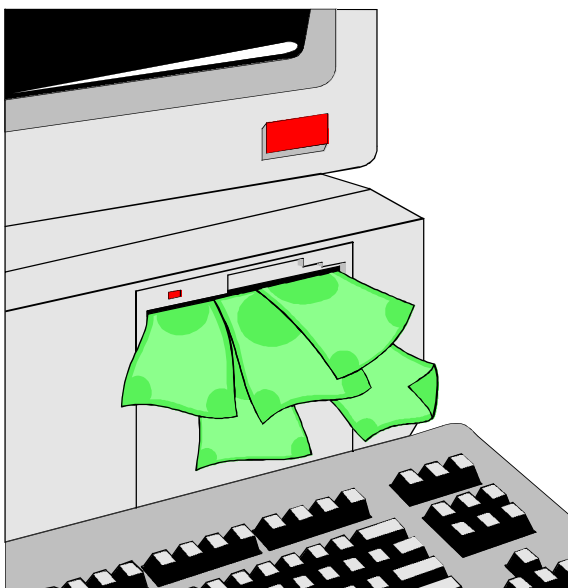
**HBCI - Homebanking Computer Interface**

**- HBCI Compendium -**

going into the new world of homebanking

---

Version 1.1 / April 1997



## Contents

1 GENERAL DESCRIPTION.....	3
2 SYNTAX.....	4
2.1 Fonts.....	4
2.2 Delimiter Syntax.....	5
2.3 syntax elements.....	5
2.4 message structure.....	7
3 DIALOG FLOW.....	7
3.1 dialog initialization.....	7
3.2 return codes.....	10
3.3 status protocol.....	10
3.4 synchronous / asynchronous processing.....	11
4 BANK PARAMETER DATA (BPD).....	11
4.1 HBCI parameter data in general.....	11
4.2 structure of the BPD.....	13
5 USER PARAMETER DATA (UPD).....	13
5.1 structure of the UPD.....	13
6 SECURITY.....	13
6.1 security aspects.....	13
6.1.1 security algorithms.....	13
6.1.2 authorization.....	15
6.1.3 authentication.....	15
6.1.4 proof of the origin.....	15
6.1.5 integrity, electronic signature.....	15
6.1.6 secrecy - ciphering.....	15
6.1.7 validity - "replay attacks".....	16
6.1.8 security media.....	16
6.2 HBCI and security.....	16
6.2.1 multiple signatures / off line processing.....	17
6.2.2 key management.....	17
7 BUSINESS CASES.....	17
7.1 single credit note / single debit note.....	17
7.2 collective credit note / collective debit note.....	17
7.3 balance inquiry.....	17
7.4 sales statistics.....	18
7.5 outlook, further business cases in planning.....	18
8 APPENDIX.....	18
8.1 specification of transport services.....	18
8.1.1 T-Online with ETSI 300 072 ("CEPT") / EHKP / BtxFIE.....	19
8.1.2 T-Online with X.28 (VT100) / X.29 / TCP/IP.....	19
8.1.3 TCP/IP.....	20
8.1.4 other transport services.....	20
8.2 Communication access.....	20
8.3 chipcard application.....	21
9 HBCI - CUSTOMER SYSTEMS.....	21
9.1 financial management software.....	21
9.2 Intelligent WWW-browsers, KIT-decoders or proprietary systems.....	22
9.3 SmartPhone.....	23
9.4 other customer systems.....	23
10 POSITIONING AND OUTLOOK.....	23
10.1 SET.....	23
10.2 OFX.....	23
10.3 CyberCash, electronic commerce.....	24
10.4 Outlook.....	24

## **Preface**

The HBCI compendium shall give a summary of the new homebanking standard and provide a classification within the world of online-services. HBCI is defined technically in a version 1.0, a corresponding agreement of the banking organizations, in this case the Central Finance Committee (ZKA), was signed in October 1996.

The structure of the HBCI compendium is approximately like the HBCI specification V1.0 . In addition some remarks of the authors are at the end of this publication, to classify HBCI within the world of other actual or planned transmission standards in the financial branch.

Additional information and a special part with an actual FAQ section are placed on the internet WWW address:

<http://members.aol.com/sxsigma/index.htm>

## Actual Situation

In the beginning, we want to explain some aspects to the definition of the abstract "homebanking".

In demarcation to *Phone-Banking*, where the transactions are communicated by voice, PCs or other intelligent terminal equipment (e.g. SmartPhone) are used for home banking. From the application view homebanking not only means information inquiry but also transaction processing.

In this moment, homebanking exclusively works with T-Online - the engagement of some Internet pioneers neglected. In case of T-Online we must distinguish two variants:

- **Screen-Dialog**

Screen dialog, like the name already says, is based on an online dialog of Btx (the German „ScreenText“ provided by Deutsche Telekom since the beginning of the '80) frames (CEPT), where 24 x 40 characters can be represented on a display. Adapted transaction forms are filled out by the customer (or a macro) and sent to the bank. The whole process takes place in the closed T-Online environment. To the protection of the banking dialog at session start a Personal Identification Number (PIN) is sent. Every banking transaction is protected by an one-time transaction number (TAN) in addition. Transaction numbers are communicated to the customer in form of TAN lists by letter post. The management of these lists on the customer- and banking-side is very time consuming and fault-prone.

Remark: Theoretical security problems consisting of monitoring telephone lines and modifying banking orders are not in the scope of this document.

- **ZKA Standard**

This "standard" became operative in 1987, when the "Central Finance Committee (ZKA)" tried to solve the compatibility problems in the homebanking area. With "ZKA Dialog" the content of a business case is sent in a net data format in a logically compressed form between the customer and the credit institute, but still based on CEPT frames. Thus unfortunately works with T-Online only on the first hand, and on the other hand does not use the optimizing methods ("Transparent Data") of T-Online either. In addition the standardization is not strong enough.. So, there exists a lot of "dialogues", which makes normalized communication impossible.

Result from this short consideration is, that there exists de facto no standard of home banking at present which is the main reason for the establishment of this new standard, named HBCI.

A second essential reason is the commercialization of the Internet, which has led to a fast development within the last two years. This platform is of highest interest in all credit institutes, not only in the area of direct banking orders but also at the general payment transactions (keyword: Electronic Commerce). The missing of generally usable security mechanisms, underlines the importance of this new banking standard HBCI also in this area.

# 1 General Description

**HBCI** is a new standard for the communication between intelligent customer systems and the corresponding computing centers for the exchange of homebanking transactions. The transmission of data is done by a net data interface, which is based on a flexible delimiter syntax (similar to UN/EDIFACT).

Target group for HBCI are the private customers and the market of small and middle sized companies; this one at the moment is still dominated by the existing T-Online applications.

There are several reasons for the engagement of the banking business in HBCI. The existing CEPT applications secured with the outdated PIN/TAN security mechanism cannot be used in future because of missing user-friendliness and presentation, additionally sufficient security mechanisms are still missing in the Internet. Still other aspects come up here, e.g. the charging policy of the German Telekom at the telephone networks and the growing importance of intelligent customer systems in the area of the private financial administration, which requires other techniques as the "ZKA-Dialog" can provide.

**All of these aspects lead to the following requirement list for a new home banking standard:**

- The data interface must be very powerful and flexible.
- It must be independent of presentation services.
- It will be supposed to transfer net data, to minimize amount of data and costs. The preparation of the transactions has to be done in the terminal equipment.
- The data interface must be independent of the transport network and so be suitable for alternate networks, e.g. the Internet or Pay-TV Networks. Underlying transport layers must be defined exactly to be able to produce banking independent customer systems with a common access protocol.
- Extended security functions shall facilitate the operating in unsafe networks and increase the consumer convenience.
- The complete solution shall be bank independent to be able to manage all accounts with the same mechanisms. It shall be manufacturer-independent to avoid compatibility problems of mobile terminal equipment, e.g. in hotel lobbies.
- HBCI shall help to enlarge the attractiveness of homebanking by providing more lines of access and business cases.
- The standard shall also be used with other banking applications e.g. in the self-service area, to reach the consumer with identical functionality over different service channels. Additionally, this improves the development and maintenance of new applications, because there is only one process necessary.

With the creation of the HBCI standard the German banking business wants to ensure that manufacturers have a long-term planning possibility for the design of customer friendly homebanking programs and systems.

The criteria mentioned above are discussed in the following chapters.

## **2 Syntax**

### **2.1 Fonts**

HBCI uses the ISO font 8859, in which the code set particularly used can be defined. Within the code set, e.g. "latin", a national, HBCI-specific subset, can be selected additionally. At this moment German, English and French languages are defined. Further definitions can follow depending on necessity.

This font applies to all document formats, but not to binary or transparent data. "Binary" means all kinds of binary programs, multimedia data, a.s.o., "transparent" means all non-HBCI-formats, usually from the banking business area (e.g. DTAUS, S.W.I.F.T.). Binary and transparent data need the full 8 bit extent of a byte, thus HBCI requires a transparent transmission network. If the transport protocol does not provide it (e.g. in case of core SMTP), filters like UUENCODE or MIME have to be used. It is important to understand that HBCI isn't text-oriented. Therefore it can be used independently of presentation services like CEPT or HTML.

## 2.2 Delimiter Syntax

HBCI uses his own delimiter syntax for the representation of data. The definition has followed UN/EDIFACT, however the formats are built up with other rules. This lies on one hand at the complexity of the UN/EDIFACT formats and on the other hand at the amount of different business cases expected in the area of private customers, which aren't defined in UN/EDIFACT until now.

A delimiter syntax has the advantage of flexibility and minimizing of data volume. Data fields are transmitted in their actual length only. Descriptive information like 'field name' and 'section length' are implicitly contained in the respective segment definition and therefore aren't sent. An additional optimization is achieved by logical compression: Optional sections are located at the end of a data structure and so that they can be cut off easily. Truncating of data items is possible too with the HBCI delimiter syntax. The following delimiters are used:

<b>+</b>	Data item end
<b>:</b>	Group data item end
<b>'</b>	Segment end
<b>?</b>	Cancellation character (to skip control characters in the text)
<b>@</b>	Switching code for binary data

The structure of the formats follows various rules. The contents of the security segments are taken from UN/EDIFACT. Suitable formats from the banking business (DTAUS (a german format for the company business), S.W.I.F.T.) are used, where standardized routines are necessary. So-called 'HBCI specific formats' cover the range, which wasn't standardized yet in any committee. In addition, these offer the possibility to form new bank-independent business cases. The bank-independence is reached by the following aspect too: a minimum number of data items, which is necessary for the smooth processing of a business case, are defined as required fields, while for information data items, which cannot be processed by all institutes, are optional fields used. Because of this, the bank-independent definition is possible on one hand, but on the other hand the flexibility is not limited.

## 2.3 syntax elements

The HBCI delimiter syntax offers 3 logical structures:

### 1. data items

Data items represent the single sections of a segment. In the simplest case e.g. one "bank code" is shown by a data item (DE). In the extreme case, a full S.W.I.F.T. format hides itself behind one data item transparently. Data items (and also groups of data items) have no administrative overhead in form of a header. The description of the features is described by their position within a segment.

### 2. groups of data items

Logically matching data items are summarized to groups of data items

(DEG). The contained elements then are described as group data items (GD).

3. **segments**

All logically matching data items and groups of data items are summarized to a HBCI segment. In the banking sense a segment generally represents a business case, e.g. a single credit note. A segment is described by an administrative supplement, the *segment header*, unlike the hierarchies of DE and DEG. It contains the unique *segment reference* primarily, which makes it possible to point exactly to any segment and to any included data or group of data item. Thus the segment reference "HKUEB" describes the customer message for the single credit note inclusive the attributes of all data items contained. E.g. the 'customer account' data item is defined in the following way: 'max.. 30 characters', 'alphanumeric', 'required field'.



## 2.4 message structure

A segment can cover the banking part of one business case only. The combination of several segments forms one **HBCI-message** which can be transferred in form of a *customer- or bank-message* as an isolated unit. Several equal-type business case segments can be within one message, e.g. 5 single credit notes. A message generally has the following structure:

- **message header**
  - **signature header**
    - **business case segment 1**
    - **business case segment n**
  - **signature trailer**
- **message trailer**

Optionally there exists a *ciphering-header* for the ciphering of data and eventually more additional *signature-headers and -trailers* for multiple signatures. The *signature-header and -trailer* of a *bank-message* is optional anyway.

The message header contains administrative information, e.g. a *message number* and a *reference number* for customer- and bank-messages. The message-trailer references to the message-header. This mechanism is taken from UN/EDIFACT.

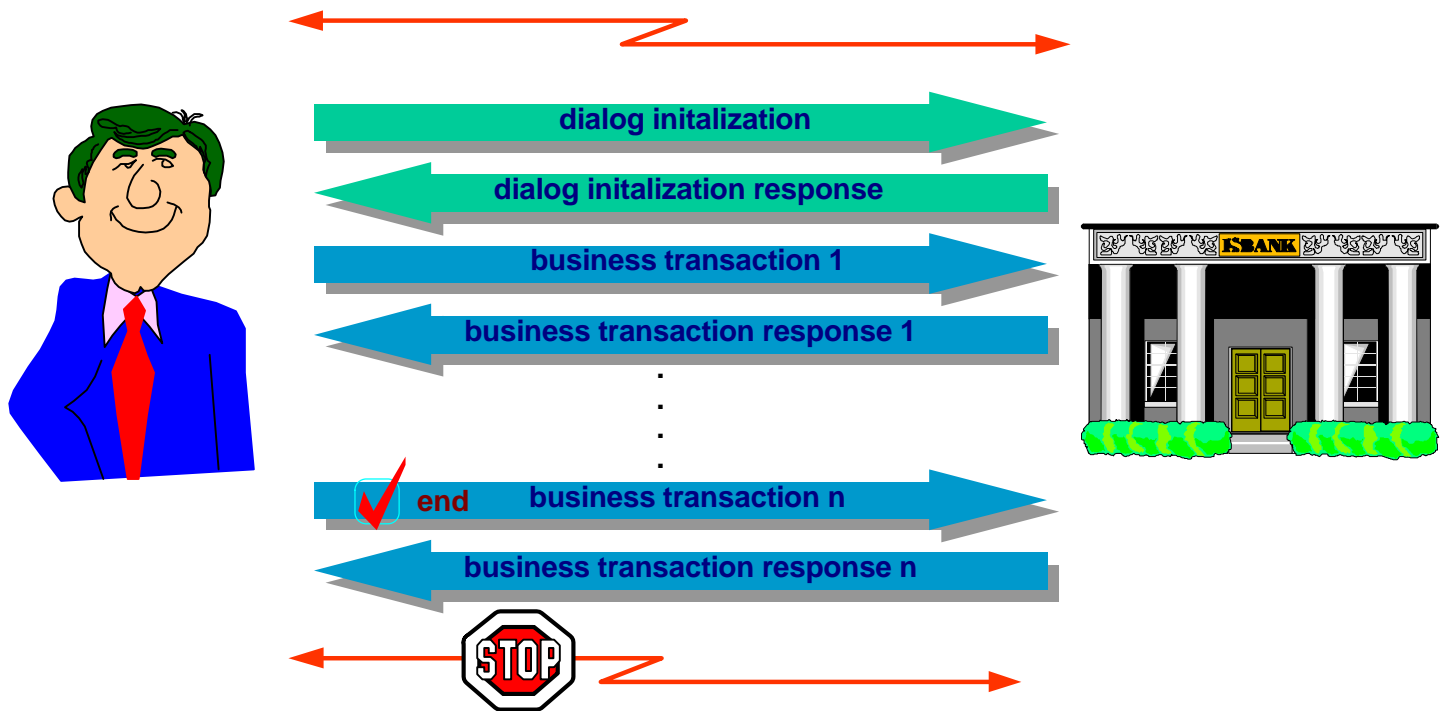
The signature-header contains information about the security mechanisms to be used and additionally one unique reference-number for the validity control at the banks side. In the signature-trailer the electronic signature for the complete message is located.

The business case segments of the bank-messages contain references to the customer-segments and in addition one return code per segment, which makes it possible to get information about the processing state.

## 3 Dialog Flow

### 3.1 dialog initialization

HBCI messages are isolated processing units, but however they can be sent only within a HBCI-Dialog (this forbids itself the use of HBCI with E-mail-services). A HBCI-dialog is built up as follows:



- **"dialog initialization" - customer-message "**
- **"dialog-initialization" - bank-message "**
  
- **1<sub>st</sub> customer message with transaction segments**
- **1st answer of the credit institute**
  
- **...**
- **...**
  
- **n<sub>th</sub> customer message with transaction segments and attribute "end of dialog" in the message header**
- **n<sub>th</sub> answer of the credit institute with attribute "end of dialog" in the message header**

The presented flow shows that a HBCI dialog is synchronous. This means, that every single message must be answered by the credit institute before a new message can be sent.

The dialog initialization ensures the mutual authentication of the two partners (customer and bank) to process the following messages which contain the order segments in a secure environment. Additionally, within the context of the dialog initialization the ciphering- and compression-methods are negotiated and the versions of the bank parameter data (BPD) and user parameter data (UPD) are adjusted. If necessary, new versions of BPD or UPD are transferred within the corresponding bank message. A similar process applies to the comparison of the versions of the public keys of the credit institute.

In the bank message also *customer specific news* can be contained, e.g. **"your new ec-card is available"**.

As a special service, a so-called "anonymous logon" is possible with a bank code as selection criteria. By use of this way a customer can get the actual BPD of the corresponding institute at least, to get information about the supported transaction types (which represents business cases) of that bank.

### 3.2 return codes

In the context of a bank-message standardized return codes are transferred (bank independence!). This code can identify a faulty data item exactly on base of the reference information in the message- and segment-header and allows an intelligent reaction of the customer system due to the meaning of that specific return code. For example one customer system can offer a bank code search mechanism to the user after recognizing a faulty receiver's bank code (return code 9210), to correct the corresponding order. The return codes are classified into error classes, which makes a detailed reaction of the customer system possible. Error reaction tables make the implementation easier. The possibility of sending individual text in addition, helps non-PC products (e.g. Smartphones) to show at least this information to the customer.

### 3.3 status protocol

In contrast to the conventional homebanking solutions, where after any interruption the state of the transmitted orders is unknown, HBCI offers two possibilities to synchronize with the banking side:

1. **dialog initialization with synchronization**

with this method, the dialog initialization message has a synchronization segment inserted, which returns the message number of the last correct message.

2. **status protocol**

in a special segment of the type "status protocol" the customer will be informed about the status of his last order in form of corresponding return codes.

This status protocol can inform a customer about the progress of his orders at normal processing conditions without error situations too. In this case a business case can change his state according to the progress, i.e. from "order accepted" to "order committed". These states are bank-dependent.

By the use of status protocols, synchronous and asynchronous processing on the banking side is also possible.

### 3.4 synchronous / asynchronous processing

Although HBCI as a data interface specification does not distinguish any different modes of operation, it shall be pointed to a possibility due to the use of status protocols:

#### **synchronous processing:**

The processing of a message is done as known from the existing T-Online dialog: One transaction message is interpreted by the banking computer, the single orders are sent to post-connected booking systems and after a reaction of these systems, an answering message is sent back to the customer. This method is also necessary with HBCI to process so-called "Inquiry-Orders", i.e. for requesting sales statistics and balance information. The disadvantage of this method is, that a physical connection must exist during the complete process.

#### **asynchronous processing:**

With this method the customer receives only the receipt of the message ("order accepted"), whereupon the customer system can interrupt the physical connection. The processing in the banking computer is executed off-line. By use of status protocols the customer can be informed by his banking software about the progress of his orders later.

## 4 Bank Parameter Data (BPD)

### 4.1 HBCI parameter data in general

At the layout of bank-independent systems limits exist, which are caused by the various system architectures of the different credit institutes. A simple example is the number of 'use lines' within a single credit order form. With the DTA format generally 14 lines are intended for the use; depending on the institute however 1 to approx. 3 lines are interpreted. These differences must be eliminated by a parameter function, so that a customer can use the same logic for all his banking connections and accounts. The customer system must be able to convert the parameter function correspondingly. Such restrictions are contained in the BPD.

Furthermore customer-specific differences take into account too. The dialog initialization contains an userid, which identifies the customer (more exactly: the security medium of the customer). The assigned accounts and the permitted order types are contained in the UPD.

Generally BPD and UPD have no legal relevance, since they only inform the customer system, which possibilities, business cases and restrictions are provided on the banking side. The exams of course are executed in the banking system as well at the submission time of the orders. It is also possible, that on the banking side orders are declined, which were correct on the customer system on base of the local BPD/UPD, because the original ones have been changed in the meanwhile. This concerns the account limits of the customer primarily, which are subjected to change by the influence of some parameters. The update (substitute) of the BPD and respectively the UPD happens by means of a version control at time of dialog initialization.

The implementation of the BPD and UPD is obligatory for customer- and banking systems, because with this feature the operator convenience can be increased enormously. The obligation refers to a part of BPD / UPD data, which is defined only with required fields. The reason is, that the parameter data itself is a completely new con-

cept, which does not exist in any banking system yet. According to a positive resonance on the customer side, these restrictions will be superfluous soon.

## 4.2 structure of the BPD

With the bank parameter data the customer system will be informed about the infrastructure of the corresponding credit institute.

- A *general part* describes the general basic conditions, e.g. the exact name of the institute, the supported languages, a.s.o.
- In the segment *communication access* the available transport services are described.
- Under the topics *ciphering* and *compression*, the methods supported by the bank are listed.
- The remaining segments contain the so-called *segment type features*, in which the restrictions per business case are described. In a common area characteristic parameters are situated, e.g. the *number of authorized signatures allowed* or the *maximum number of orders per message*. The rest really contains the specific restrictions per business case type, i.e. the *number of use-lines* mentioned above.

## 5 User Parameter Data (UPD)

General remarks to the HBCI-parameter data can be found in the chapter IV: "BPD".

### 5.1 structure of the UPD

With the *user parameter data* the customer system gets information about the profile of a user, exactly about an *userid* (= security medium). With an intelligent interpretation of the UPD, the customer system is able to construct a real powerful *cash management system*, since, among other things, it receives information about *limits per account* and can optimize the cash flow.

- under *general user parameters* information like the *userid* and data for the version control can be found.
- The remaining segments are of the type *account information*, which can contain general information per account like *account numbers* and *product names* as well as the *permitted transaction types* and corresponding *account limits*.

## 6 Security

Powerful security mechanisms are the main feature of this new HBCI architecture. The basics are comparable with the "ZKA agreement for business customers". Even in the security area, many details are taken from this standard as well as from UN/EDIFACT.

To be able to classify the different functions, the first topic explains the different security aspects:

### 6.1 security aspects

#### 6.1.1 security algorithms

Basically two methods exist at HBCI, which are described in detail later. At this point only a brief description follows for the understanding of the context:

- **DEA (Data Encryption Algorithm)**  
DEA, also known as DES (Data Encryption Standard), is a symmetrical method. This means, that keys, which are used for signing or encoding,

must be known by both partners. Therefore, they must have been communicated before over an alternate path. This common known key is a secret key, since the security depends upon the fact, that only the two parties involved know this key. With HBCI and DES two key types are used, namely one *signature key* to 'sign' the messages and one *encryption key* to cipher messages by means of dynamic *message encryption keys*.

*This DES-algorithm at HBCI version 1.0 is supported by the use of the ZKA chipcard.*

- **RSA (Rivest - Shamir - Adleman, named after the inventors)**  
With the asymmetrical RSA-method key pairs are used, always consisting of one *private key* and one *public key*. The idea is, that a customer creates one pair of keys by software and signs his orders with his private key. The credit institute can check the electronic signature by means of the public key on correctness. The public key proves the origin of the signature unequivocal. It must not be kept secret, because with this key signatures only can be checked, not be produced however. In HBCI two pairs of RSA keys are used, namely one *signature key-pair* for signing messages and one *encryption key-pair* for the creation of dynamic *message encryption keys*, in case of ciphering messages.  
*The RSA-algorithm in HBCI version 1.0 is supported by a software solution with diskettes or fixed disks as storage media.*

After this discussion about security mechanisms now to the real aspects of the security:



### **6.1.2 authorization**

Authorization in this context means the userid-check against the security medium. During the authorization the user is asked to issue a password or PIN before the processing of any security functions can take place. The password is checked locally (it does not leave the customer system). At DES this check is executed within the chip-card, at RSA in the PC software of the customer system.

### **6.1.3 authentication**

With the mutual authentication the two communicating parties make themselves confessed to each other. This happens during the dialog initialization by signing the customer- and bank-message respectively. If the signature of the partner can be verified successfully, this examination is completed. By use of RSA only a one-side authentication of the customer is processed optionally.

### **6.1.4 proof of the origin**

At submitted orders it is important, that the origin of the message can be proved obviously ("non repudiation of origin"). This is only possible with the *electronic signature* of RSA, because here the customer is identified by his unique private key. With DES it depends on the trustworthiness of the partner, because he also knows the key.

### **6.1.5 integrity, electronic signature**

The electronic signature shall prove, that a HBCI-message wasn't modified in any way on the transmission path. To do this, at first a hash value is calculated - a kind of a cryptographic checksum - over the complete message. With the result of this process an electronic signature is calculated in accordance with DES or RSA, which will be put into the HBCI segment "signature trailer".

The receiver forms this hash value with the same algorithm and checks the signature by means of the secret key (DES) or the public key (RSA) respectively.

### **6.1.6 secrecy - ciphering**

Unlike the electronic signature, where the message is readable, with ciphering the complete message is scrambled and therefore made illegible. This is necessary for the transmission of confidential information, e.g. of sales statistics. An encoded message without signature can be changed on the transmission path, the changes shall have no consequences, since the message isn't readable however. Only a combination of both methods therefore can create the desired result.

HBCI uses the triple-DES algorithm for the ciphering of data generally. Instead of the static encryption key or encryption key-pair respectively, a dynamic message-specific key is used for the encryption. This consists of a random number, which is encoded with the encryption key and inserted in front of the encrypted message.

### 6.1.7 validity - "replay attacks"

One of the possible attacks in a cryptographic system is described in the following scenario:

The attacker monitors a telephone line, stores the information of a dialog and replays the stored information repeatedly ("replay attack"). With this method, the attacker cannot attain any financial advantage, but however he can annoy the customer concerned.

As a criterion for the unambiguity a time stamp is used often, what on one hand is not supported by all of the HBCI-terminals (e.g. SmartPhone), on the other hand does not work reliably, since it cannot be ensured, that every PC clock supplies the timestamp with plausible results.

Therefore, within HBCI another method for proving the validity was defined, which works in a safe way and is not too complex to handle. It consists of a combination of a sequence counter, which is stored on the security medium and the banking system in parallel, and a list of already submitted sequence numbers over a particular time period. This second part is necessary, because of the existence of offline transactions. In this case it is not sure, that orders, which are created without a physical connection by different persons at different times, have ascending sequence numbers, when they arrive at the banking system..

### 6.1.8 security media

The target system is a RSA-based solution with a ZKA-certified chipcard. In this moment RSA chipcards are too expensive and too slow, therefore HBCI version 1.0 defines two security mechanisms. They are software-compatible as far as possible, to ensure an easy migration, when the target system will be available.

#### DES with a ZKA chipcard

The DES-method has the disadvantage of the possibility of the repudiation of the origin. This is not a real problem, because of the trust between the customer and his bank. Otherwise this method has of course the advantage of a chipcard as a hardware medium. All sensitive cryptographic processes run within the chip and are not accessible from the outside. Additionally, the chipcard security is mobile. So this solution can be used also in an unknown environment (e.g. a public HBCI customer system in a hotel lobby).

As security medium a ZKA chipcard is used, identical to the ec-chipcard. This type of card also was tested in the field test for the electronic purse in Ravensburg/Weingarten in 1996. Now 25 million chipcards are given to the customers.

#### RSA software solution

RSA is the target system for HBCI security. In the HBCI specification version 1.0 all of the cryptographic functions are executed within the main storage of the terminal. One advantage of this method is, that RSA homebanking is possible without additional hardware investments (chipcard reader). The mobility can be reached by use of diskettes within a **secure** environment.

## 6.2 HBCI and security

All methods and mechanisms listed above are described in the HBCI specification in detail. The implementation is mandatory for customer- and banking systems for the

most parts. Optional features are contained in the area of ciphering and in the signing of the bank messages.

### **6.2.1 multiple signatures / off line processing**

Because HBCI affects not only private customers, but also small companies, scenarios with multiple partners must also be practicable. The following scenario must be covered by use of HBCI-procedures: A secretary writes orders and saves them on a server. These orders are signed by the managing director and a signing clerk later. After this the secretary sends all orders of the day in one dialog to the banking system. An effective validity control, exactly defined in HBCI, prevents from crediting orders twice, because of line- or processing problems. Such processed can be executed with different security media of the same type, which naturally must have valid account authorizations on the banking side.

### **6.2.2 key management**

There exist special transaction types for changing and locking keys. The changes concern only the RSA-method, since the keys within the chipcard are fixed.

## **7 Business Cases**

Version 1.0 of HBCI defines only the classic business cases, to build up the necessary infrastructure at the customer- and banking-side first. It is planned however, to use HBCI as a secure container for the information flow over different transport media. More aspects you'll find in the 'Outlook' section of this publication.

### **7.1 single credit note / single debit note**

"Single Credit/Debit Notes" are defined as HBCI specific formats to ensure compatibility with the planned transaction types "Scheduled Credit Note" and "Standing Order". The accounts for senders and receivers are interpreted as standardized groups of data elements. The number of use-lines is adjustable per institute with help of the BPD. Special rules are defined for particular forms of the credit note, i.e. donations or the use with a check digit.

### **7.2 collective credit note / collective debit note**

For the "Collective Credit/Debit Note" the DTAUS format is used transparently. This is caused by already available processing systems in the credit institutes, which also handle orders from other sources (e.g. DTA diskettes).

Collective credit notes are a typical example for asynchronous processing. Here it is suggestive, especially with bigger-sized orders, to accept and confirm them online, but process them without any line connection afterwards. The progress can be monitored by the use of status protocols..

### **7.3 balance inquiry**

The "Balance Inquiry" transaction type is defined as a HBCI specific format. Commonly used balance values are provided and optionally information about limits too. The balance inquiry is a prototype for a so-called "Inquiry Order". It is either possible to issue a specific account number or the information "all accounts" to get the desired data. With the second possibility, the balances of all available accounts for that customer are supplied.

## 7.4 sales statistics

"Sales Statistics" are requested also via an inquiry order. As response, booked sales or respectively not credited sales are sent via MT940 and MT942 formats. If it is necessary to send big amounts of data, the use of the data item "rerun point" is possible. This means, that the customer system gets the information, that more data is still available. So it can set up a new inquiry order with the rerun point enclosed, to get the next part of the information. The data item "Maximum Number" can be used to restrict the amount of data, so that terminals with a low screen resolution and memory size (e.g. Smartphones) are not overcharged.

## 7.5 outlook, further business cases in planning

In addition to the described business cases, the following order types are already planned:

- scheduled transfers, standing orders
- loading of the stored value chipcard (personal ATM)
- financial reports
- credit card sales
- time deposit
- foreign countries transfers
- check orders, communication to the bank
- stock exchange quotations
- exchange rates
- investment
- financial calculations
- marketing and service offers of the credit institutes
- filetransfer

In addition to the order types already shown, you can imagine, which further possibilities exist with such a powerful tool. It begins with banking offers over multimedia techniques, continues with cyber shopping in open networks with chipcard security and ends with the possibility of general data transfer via secure "HBCI-Containers".

# 8 Appendix

## 8.1 specification of transport services

In the previous chapters, numerous aspects of the flexibility of HBCI were shown. In this chapter the philosophy is exactly vice versa. The major purpose of the definition of transport services in conjunction with HBCI is, to fix the different possibilities as exact and restrictive as possible, so that two partners immediately can communicate without further coordination. In this version of HBCI, three different protocol stacks are defined. Two communication channels are in the scope of T-Online, a 3rd one is intended for the use of intelligent HBCI-clients connected over TCP/IP (Internet).

### **8.1.1 T-Online with ETSI 300 072 ("CEPT") / EHKP / BtxFIF**

The classic homebanking applications work over CEPT and EHKP presently. Therefore both of the possible connection types (EHKP and X.29) are defined. In the HBCI context, CEPT and EHKP are used only as transport frames for transparent data. Since there exists an EHKP restriction due to the maximum size of a dialog field of approx. 1600 bytes, the "Btx File Interchange Format" (BtxFIF) is defined as chaining protocol additionally. A chapter of the specification describes all parameters necessary for CEPT, EHKP and BtxFIF in detail. Certain optional functions of BtxFIF, how e.g. the restart facility, consciously are not used, to make the design of customer systems more simple.

### **8.1.2 T-Online with X.28 (VT100) / X.29 / TCP/IP**

Unlike EHKP, the connection via the so-called "Triple-X-Protocols" (X. 28, X.3, X.29) is still unusual in the homebanking area. The advantage of this solution lies on one hand in the low overhead regarding the transparent data transmission, on the other hand in the internationality of this standard. The main problem of X.29 is the missing end-to-end flow control, which makes the use of a separate TCP layer necessary. Therefore HBCI defines X.29 in combination with TCP/IP as the second T-Online protocol stack.

### **8.1.3 TCP/IP**

This definition of the TCP/IP protocol stack is the general form of the description under topic 8.1.2. All well known SLIP- and PPP-connections are supported here. As a standardized interface to the application layer, the TCP socket-interface is used, consisting of IP address and port number (as HBCI port number, "3000" is registered at the responsible internet committee - IANA). It is important to know, that in HBCI version 1.0 only this bare TCP/IP connection is specified. One reason is, that actually no WWW-browser is able to handle HBCI messages. This will change basically in the next few months because of the use of intelligent program additions like "Plug-Ins" or "Java-Applets". If this is foreseeable (and also technically practicable), corresponding HTTP-, HTML- or other definitions will follow soon.

### **8.1.4 other transport services**

In addition to the above protocols, depending on need, also other transport services are to be defined in HBCI. Actually this is not necessary, because alternate network providers like CompuServe, AOL etc. also change their engagement in the direction of internet. So the transport services defined until now (and in the near future) are sufficient at the moment. The next enhancement can be done in the direction to digital TV.

## **8.2 Communication access**

A great problem at the present homebanking support is the manner of the communication access. In T-Online actually one gateway frame is used per bank, mostly even per branch office. Therefore hundreds of gateway frames are needed to select the right institute (in a few dozen of computing centers). Therefore in HBCI the bank code is sent along with the gateway frame, with the consequence, that only one gateway frame is needed per banking system. With this solution, the problem only is defused, but the instability because of addressing changes still exists. Therefore a "net access database" was defined, consisting of all access parameters, necessary to communicate with a specific credit institute. The entries in this database are coded in HBCI-syntax and can be requested with a special inquiry order. The trick is, that there exist only four fixed addresses per network to obtain an actual data base entry, namely one per association (private banks, savings banks, cooperative banks, public banks). With this method the manufacturers of customer systems hopefully will get a better service to update their access procedures.

### 8.3 chipcard application

With the symmetrical method a chipcard according to the ZKA-specification is used. It works with the same operating system as the new ec-chipcard, which is also used for the electronic purse. The homebanking extension consists only of some additional fields (no special commands!) within this chipcard to cover the following items:

- the use of an own Banking-PIN, kept separately from the general ec-PIN.
- customer individual sign- and encryption keys.
- a chipcard field to save the sequence number, used for the validity control of the orders.
- several banking connections (approx. 3 to 5) with the following structure:
  - nickname of the bank
  - bank code and userid
  - net access ID (at T-Online: gateway frame number)

This description shows, that the chipcard is bank independent and also suitable for mobile use because of its stored net access data.

## 9 HBCI - Customer Systems

There are generally different kinds of customer terminals, coming in question for the use with HBCI. These shall be described now briefly:

### 9.1 financial management software

It is to expect, that special financial management programs will be the first prototypes supporting HBCI. Already today these systems like Quicken or MS-Money offer the possibility of sending orders via screen- or ZKA-dialog to the corresponding credit institutes. HBCI in this class of terminals has primarily the following advantages:

- effective and fast transmission of net data.
- independence of the transport service.
- automatic configurability with the help of UPD and BPD
- increased comfort and functionality in the area of security.
- new business cases, which challenge the creativity of manufacturers of customer systems.
- bank independence.
- no problems with the maintenance of bank accesses.

The list will be enlarged in the short-coming, but the main aspects of HBCI are recognizable. At this point it has to be remarked, that some banks or banking groups surely will promote HBCI products with their own functionality and presentation, suitable for the use as their own marketing and service instruments.

## **9.2 Intelligent WWW-browsers, KIT-decoders or proprietary systems**

This section describes the second step in the HBCI-development. The actual discussion and the enormous progress in the development of WWW-browsers gives an impression of products of the near future, which also will be able to generate and handle HBCI-messages. Open is in the moment, which standard will be the appropriate for such operations. Though Java and applet techniques are state of the art actually, but the handling of HBCI messages is rather complicated. Particularly the communication with a chipcard must be possible (medium-term anyway), what doesn't make the thing easier. Nevertheless this class of HBCI terminals will be able to play an important role in the market of the near future. At the Cebit '97 exhibition we presented HBCI home-banking with chipcards by use of Java applets.



### 9.3 SmartPhone

The concept of "homebanking" generally implies the use of a PC. Notwithstanding the wretched attempts of the past to place telephones as terminals in the market in remarkable amounts, HBCI opens up new ways here. The main argument for this class of devices is surely the increased security comfort, primarily caused by the possibilities of the chipcard. Generally no diskette solutions are to be expected in this segment. For a target group, who has certain fears of contact in the working with computers, SmartPhones are a considerable alternative. This also applies to people, who are forced by their job to take care of her banking transactions in transit.

### 9.4 other customer systems

Before the appearance of a new communication standard it is difficult to make statements about its future progress. But the rapid development in the client market, caused by the even faster development in the telecommunication area opens up possibilities for the use of HBCI completely different from the actual homebanking situation. So an integration of HBCI into so-called "Set Top boxes" is possible, just like into the new genus of the internet-PC. The decision over the success of HBCI surely will fall in the internet area and the corresponding kinds of applications.

## 10 Positioning and Outlook

In relation to HBCI we must take a look to the following current developments:

- **SET - Secure Electronic Transaction (Visa and MasterCard)**
- **OFX - Open Financial Exchange ( Microsoft, Intuit and Checkfree )**
  - **SSL - Secure Socket Layer ( Netscape (C) )**
- **CyberCash**

### 10.1 SET

A comparison of HBCI with SET is quite simple. Visa and MasterCard (and involved companies, e.g. GTE, IBM, Microsoft and Netscape) have defined SET for the general handling of the credit card business. This starts with secure payment in the area of CyberShopping and ends with the administration of the credit card accounts. Although there are certain coincidences in the traditional banking area, the concept of this standard is adapted at the American market and is not applicable to the German situation. It is however remarkable, that the outstanding security mechanisms under inclusion of certification authorities show very strong similarities with HBCI. As transport service for SET TCP/IP is described.

### 10.2 OFX

The positioning of the new standard OFX is quite different. The orientation towards the American market is transferable from SET. But in difference, OFX exclusively uses HTTP as a transport protocol and SSL for the data protection. So OFX is not comparable with the net data concept of HBCI. The data here is sent over the line in special document formats similar to HTML. The only defined transmission medium is internet in combination with WWW surroundings. Also the security area is not comparable, because OFX uses the transport layer in contrary to the application-oriented end-to-end-security in HBCI. With OFX Microsoft, Intuit and Checkfree documents the orientation towards internet, which is surely a very good attempt. OFX is a proper solution for the

processing of banking transactions in the internet area. The difference consist only of the trade-specific (german) orientation of HBCI and the high requirements of the banking organizations on this standard, which are described at the beginning of this compendium.

### **10.3 CyberCash, electronic commerce**

The first wave of excitement regarding new protocols and arising standards in the area of CyberCash and Electronic Commerce is actually over; the articles in the professional journals are more seldom. One reason for that is surely, that corresponding methods are not established in Germany actually. What sense makes the best standard, if one needs an account in the USA for shopping in the internet malls? The topic however is still brand actual! Also in Germany new developments are growing to ensure a secure handling of payment transactions and make so CyberShopping possible. What isn't foreseeable is the effect of the different electronic purse chipcards in this area. At last we must wait, which role HBCI can play, just in connection with the medium-term available RSA chipcard.

### **10.4 Outlook**

To dare a forecast about the expecting acceptance of HBCI at present, is very critical. Remarkable is anyway the common attitude of the german banking business towards HBCI. An enormous interest in the market also results from this. Important for the success of HBCI anyway is, how fast the standard, including the new transaction types, becomes available and how fast the first attractive products will appear, which make a customers investment profitable. The success on the other hand hangs also at the security area and thus especially at the availability of ZKA RSA chipcards (and corresponding chipcard readers at the customer side). At last HBCI must be accepted as a national standard also in the international, especially in the European field, in which similar developments are arising. It is planned however, to bring in HBCI into international committees for standardization.

## Disclaimer

You may distribute this Postscript-code freely and combine it with other documents. We think however it is fair to require, that the distribution of the HBCI-compendium must be cost-free and only as a whole inclusive my address.

Please take into account, that this **HBCI-Compendium** is a protected document according to the **"copyright" (©)** rules !

You can reach us on:

Kurt Haubner:


Email: khaubner@compuserve.com

Phone: +49-(0)89-713199

Hans-Bernhard Beykirch:

Email: beykirch@compuserve.com

Phone +49-(0)228-4495-674

 IZ Computer Science Center of the German Savings Banks  
Königswintererstr. 552  
53227 Bonn  
Germany

Version 1.1: April 1997